Technical Research Note 215

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

Frederick W Kuehl

STATISTICAL RESEARCH AND ANALYSIS DIVISION

U. S. Army
Behavioral Science Research Laboratory

September 1969

Technical Research Note 215

AD

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

Frederick W. Kuehl

STATISTICAL RESEARCH AND ANALYSIS DIVISION
Cecil D. Johnson, Chief

U. S. ARMY BEHAVIORAL SCIENCE RESEARCH LABORATORY

Office, Chief of Research and Development
Department of the Army

Room 239, The Commonwealth Building
1320 Wilson Boulevard, Arlington, Virginia 22209

September 1969

Army Project Number
2Q065101M711

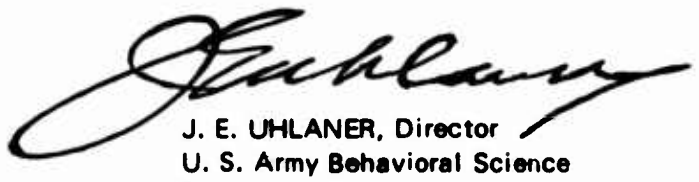
SIMPO d-12

BESRL Technical Research Reports and Technical Research Notes are intended for sponsors of R&D tasks and other research and military agencies. Any findings ready for implementation at the time of publication are presented in the latter part of the Brief. Upon completion of a major phase of the task, formal recommendations for official action normally are conveyed to appropriate military agencies by briefing or Disposition Form.

FOREWORD

The BESRL Work Unit, "Computerized Models for the Simulation of Policies and Operations of the Personnel Subsystem--SIMPO-I," is conducted by the Statistical Research and Analysis Division of BESRL. The Task constitutes the initial undertaking of an operations research requirement described in the Army Master Study Program under the title, "A Simulation Model of Personnel Operations (SIMPO)" and is Project 20065101M711, "Army Operations and Intelligence Analysis," under the auspices of the Army Study Advisory Committee. Sub-Work Units include: a) Operational Analysis of Personnel Subsystems; b) Cataloging and Integration of Existing Manpower Models; c) Development of Measures of System Effectiveness; d) Development of Modeling Techniques; e) Design and Programming of SIMPO-I; f) Application and Evaluation of Computerized Models; and g) Problem Oriented Language for Management.

The present publication reports on the development of a pseudo-random number generator which would provide random numbers for application in SIMPO-I entity models, and evaluation of the generator product through a series of statistical tests devised for the purpose. The research was conducted under the technical guidance of Richard C. Sorenson, then SIMPO Task Leader. Dr. Stanley Mulaik provided constructive suggestions on text content and derivation of the statistical tests applied. The tests have wide applicability in evaluating random number generation for computer systems.



J. E. UHLANER, Director
U. S. Army Behavioral Science
Research Laboratory

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

BRIEF

Requirement:

To develop a pseudo-random number generator which would be able to rapidly supply the SIMPO-I entity models with numbers meeting essential tests for random qualities.

Research Products:

A pseudo-random number generator was developed and tested. Repeated multiplication of a starting number by a carefully chosen constant produced a series of 47-bit binary numbers which were reduced by the modulus of the system (2^{47}). To increase randomness, the twelve low order bits were not used. The resulting numbers were converted to base 10 and assigned a decimal point to the left of the first digit. Tests were made on groups of ten numbers to evaluate their conformation to expected distributions. The generator was generally acceptable.

Perhaps equally important with the development of the generator, the work done resulted in a compilation of statistical tests which have wider application for evaluating generators used on other computer systems. The tests are described, and essential mathematical formulations are presented.

Utilization of the Research Products:

The multiplicative generator of pseudo-random numbers described here is in use at the BESRL computer installation. Tests used in evaluating this generator will be useful for evaluating the generators used by other computing centers.

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

CONTENTS

TABLES

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

Pseudo-random numbers generated by computer routines are used widely by operations research specialists and other scientists in several major problem areas.

One use is the generation of experimental samples in which each random number represents a variable for one of the entities in the sample. Personnel in the Army have been simulated, for example, by using a sequence of eleven random numbers to represent test scores for each of the eleven tests in the Army Classification Battery. The scores can be given desired distributions and interrelationship by appropriate transformation, a characteristic particularly important when evaluating a policy change which affects these aspects of the sample.

Mathematical games of strategy are another important application--one that is receiving increasing attention. A model of the system is developed which includes significant variables and appropriate interactions. By varying the relative weights and interactions of the variables, the behavior of the system under various strategies and policies is simulated, and the system is evaluated by noting the effect on criterion measures. Random numbers are used to represent the different variables.

Solution of certain mathematical problems for which a probabilistic model may be formulated is possible by using random numbers in appropriate sampling techniques. These problems include evaluating integrals, solving ordinary and partial differential equations, and working with difficult systems of linear constraints for which only a partial or "point" solution is desired.

The applications above are relevant to the simulation models developed in the Behavioral Science Research Laboratory Work Unit, "Simulation of Personnel Policies and Operations, SIMPO-I." The SIMPO-I Quality Input Model (1) and the SIMPO-I General Entity Simulator[1] both make extensive use of random numbers. The Quality Input Model has used over 200,000 such numbers in a single simulation study. Thus, an efficient random number generator is an essential tool of the SIMPO-I models.

---

[1] The product of U. S. Army Research Office contract number DA HC 1969-C-0001, with CONSAD Research Corporation, monitored by SIMPO-I personnel. BESRL report is in preparation.

BACKGROUND

A random number results from an independent trial from an infinite population of numbers whose distribution corresponds to some theoretical probability distribution function. That is, each number has an expected probable occurrence, and the outcome of each trial is a function solely of the overall distribution and is not affected by previous trials. The property of independence, which is of special significance, means that there are no predictable patterns within sequences or samples or random numbers other than those based on the laws of chance and on the theoretical probability distribution from which the numbers were drawn.

A random number generator is a method of generating numbers under the constraints of probability laws. These probability laws govern the sequence of numbers generated and describe the statistical properties of the sequence and the relationship between numbers in the sequence. Although devices to generate "pure" random numbers exist and have been used, the term random number generator normally refers to a method of obtaining pseudo-random numbers based on a simple mathematical operation that produces sequences completely determined by the parameters and initial values. Devices which generate pseudo-random numbers are normally used as the source of random numbers for several reasons. First, it is frequently desirable to have the capability of repeating exactly the sequence used in a previous problem, an impossible (or prohibitively expensive) feat with a random device. In addition, not only do efficient pseudo-random number generators exist but, if parameters are properly chosen, these generators produce numbers with almost completely acceptable properties. A generator must satisfy the following criteria to be acceptable as a source of random numbers:

1. Sequences produced must show satisfactory statistical properties.

2. Cycle length must be sufficient to insure that no more than a small portion of the total sequence will be necessary for a single application.

3. Numbers must be generated and returned quickly enough that problems requiring many numbers can be solved efficiently.

4. Storage requirements must not be excessive relative to total capacity of the computer.

Random number generators typically utilize a simple repeatable mathematical operation in which the entire sequence is perfectly determined from initial conditions. By far the most common of these have been of the linear congruential form

$$x_j = c^j x_o + \mu \ (\text{MOD } M)$$

where x_j is the random number output, c a constant multiplier, x_o the starting value, μ a constant, M the modulus of the system, and j the

position of the number in the sequence. Such generators are generally referred to as multiplicative or power residue if $\mu = 0$, and as mixed if $\mu \neq 0$. The double congruential generator may use a second generator to scramble the numbers output by the first or the output of two multiplicative generators may be added.

In spite of the considerable effort that has been devoted to developing and testing sophisticated methods of generating random numbers by using modified forms of the basic linear congruential methods, most of the literature concerned with number theory (as opposed to that concerned with specific generators) indicates that the multiplicative is as good as any other (and usually quicker). It produces sequences with good statistical properties if the parameters are properly selected and if the computer word length is sufficiently large. For example, Coveyou and MacPherson (2) conclude, "There is at present no method of generating pseudo-uniform sequences better than the simple multiplicative congruence method with a carefully chosen multiplier."

On the other hand, Marsaglia in 1968 (3) cautioned that any multiplicative generator has a defect which makes it unsuitable for certain Monte Carlo problems: Points whose coordinates are sequences of numbers generated by a single pseudo-random number generator fall in a relatively small number of parallel hyperplanes of the unit n-cube. On the basis of this fact, no single generator, whether multiplicative or congruential, should be used to generate more than one coordinate of a set of coordinates at a time; and when several generators are used simultaneously, they should be tested for independence.

The most important factor limiting the potential statistical properties of the sequences output by any generator of the congruential form is the word length of the computer. The high order, or most significant, bits of a computer word have cycles longer than those of the low order bits. Theoretically, bits with longer cycles display better random properties. Therefore, since a longer word has more bits than a shorter word, and since the high order bits of the longer word have longer cycles than those of the shorter word, it follows that longer words produce sequences that display better random properties than do the shorter ones. Both Coveyou and MacPherson strongly support this idea.

The multiplier, the other factor affecting statistical performance, can affect the statistical properties greatly, particularly for generators of relatively short word length. Coveyou and MacPherson (2) suggest the following criteria in the selection of multipliers for uniform random number generators of the congruential type:

1. The multiplier c should not be close to a simple rational multiple of M (the modulus of the system); if it is, the basic congruence shows that appreciable serial correlation will result.

2. The multiplier c should not be close to a simple rational multiple of the square root of M; even though the choice may produce very small correlation between adjacent pairs, serious difficulties result in the triplet distributio...

3. The multiplier c should not be chosen with a small number of 1's in the binary representation to facilitate "shift and add" techniques; if the number of 1's is small enough to do any good, "small" solutions of the basic congruence will again lead to trouble.

4. Above all, the multiplier c must be adequately large.

5. The choice of multiplier must be made more carefully for computers of short word length; it may be necessary to use multiple precision arithmetic for computers with short word length such as the IBM 360[2] in order to ensure unquestionably good statistical performance.

EVALUATION OF SEQUENCES OF PSEUDO-RANDOM NUMBERS

It was noted that pseudo-random numbers are not independent in the same way as random numbers. Obviously, any process in which the entire sequence is precisely determined by the parameters and the starting point does not produce independent numbers. It is not essential, however, that a generator output sequences of numbers which are literally independent; it is sufficient that the numbers be independent to the degree that a sequence or sequences do not have properties which adversely affect the results of the application for which the numbers are used. In this sense, a generator must satisfy the "independence property" and only in this sense does the "independence property" have any meaning when applied to pseudo-random numbers. To be an acceptable source of random numbers, therefore, a generator must satisfy two criteria:

1. The generator must output numbers which fit the desired distribution (in this case, the uniform distribution). This criterion is easily met. In fact, with properly selected parameters, all generators of the power residue type produce numbers which are exactly uniform over the unit interval when the entire sequence is considered. In a given application, however, only sub-sequences normally are used. More important, if every sequence were exactly uniform, the result would not be random numbers. Consequently, the concern is with various sub-sequences, which must be tested to insure that they are sufficiently uniform to satisfy the uniformity criterion yet not so uniform that they are not random. In addition, the length of the sub-sequences on which the tests are conducted must be consistent with the intended use, and the intervals (against which the distribution is tested) should be consistent with the required degree of precision.

---

[2] Commercial designations are used only for clarification of the procedures described. Their use does not constitute indorsement of the product either by the Army or by the U. S. Army Behavioral Science Research Laboratory.

2. The generator must yield sequences which do not have undesirable patterns or waves. The tests for undesirable patterns or waves within sequences is even more closely tied to the application for which the numbers are going to be used. Some applications, for example, may be affected by the distribution of sets of n numbers where n is relatively small, whereas other applications may depend on the sum of a set or the maximum number from a set. The tests used to evaluate a specific generator must be designed in consideration of the intended use in order to insure satisfactory statistical properties and reliable results.

THE NEW GENERATOR DEVELOPED BY BESRL

The generator available when the present study was undertaken used a power residue or multiplicative method in a single computer word of 24 bits. Statistical tests made on samples of numbers generated by this program indicated that the samples had unacceptable properties. Because of the results of these tests and because of theoretical considerations favoring a generator based on a longer word length, decision was made to develop a generator which used double precision arithmetic (48-bit computer words).

The new generator developed by BESRL is a power residue or multiplicative generator, the simplest of the many generators based on the linear congruential method. It has the form

$$x_j = c^j x_o \text{ (MOD M)}.$$

The j^{th} term of the sequence (beginning with x_o) is equal to the j^{th} power of c times x_o, reduced modulo M, where x_o is the starting value, c is a constant multiplier, and M is the modulus of the system.

Double precision arithmetic is used in multiplication. On the Control Data 3300 [3] computer, this means that instead of the usual 24 bits, 48 bits are used. Since the cycle length of a bit in a given position increases from low to high order bits, the 24 most significant bits using 48-bit words are theoretically more random than any of the bits when a 24-bit word is used. The increase from 24 to 48 bits is probably the most important difference between the new generator and the old. Coveyou and MacPherson, who (as mentioned previously) strongly emphasize the importance of word length as a determinate of statistical performance, are very skeptical of any generator based on a word length of less than 35 bits (2).

The new generator has a modulus of 2^{47}, which equals the word size (the 48th bit is a sign bit). A modulus equivalent to the word size avoids the necessity of division in both reducing MOD M (only the low 47 bits are

---

[3] See footnote 2 on page 4.

kept) and conversion to the unit interval (the decimal point is merely assigned to the left of the number). A modulus equal to the word size is used for most power residue generators since it also maximizes the cycle length.

Each binary number is evaluated (converted to the unit interval) based on the 35 high order bits and disregarding the 12 low order bits. This method avoids both the bias caused by restriction on the final octal position (last 3 bits) and any potential bias resulting from the relatively short cycles of the 12 low order bits.

The cycle length is 2^{45} with each of 2^{35} numbers appearing 2^{10} times. Developing generators with known adequate cycle length has not been a problem; however, the increase from 2^{21} to 2^{45} means that even for applications requiring many numbers only a tiny portion of the total sequence will be used.

The computer program has been set up to generate as many independent sequences as desired. Each sequence uses as its multiplier a constant plus i where i equals the number of the sequence times another constant. In other words, the difference between the multipliers for each pair of adjacent sequences is a constant.

On the present BESRL computer, a random floating point number is generated and returned in 120 μ seconds (i.e., approximately 8350 numbers can be generated per second).

Any odd number up to 16 octal digits can be used as the starting value (x_o). Because of the binary representation of numbers in the computer, odd starting numbers give a longer sequence of random numbers before repeating than even starting numbers--sequence length of 2^{45} as opposed to 2^{44}.

TESTS USED TO EVALUATE THE BESRL GENERATOR

Tests used to evaluate the generator include 1) standard tests designed to detect the obviously unsatisfactory generator, 2) tests developed in conjunction with a previous BESRL project to evaluate the possibility of undesirable waves or patterns within sub-sequences, and 3) additional tests designed in consideration of the use made of random numbers by BESRL in evaluating possible relationships between several independent sequences. All tests were conducted on sets of ten number and the statistics accumulated over the entire sample. For some tests, the choice represented only a programming convenience and had no effect on the numbers tested. For other tests, however, the ten-number set limitation meant that certain numbers were omitted. For example, the triplet test considers the first three sets of three from each set of ten, and the test for the minimum of seven considers the last seven of each set of ten.

In applying these tests, each sequence was treated as if it were composed of numbers resulting from a series of independent trials from an infinite population of numbers whose distribution conformed to the uniform distribution. In other words, it was assumed that the numbers met the uniform distribution requirement of random numbers. The numbers were then tested to determine if their characteristics were consistent with what would be expected for random numbers. All tests involved comparing a distribution of actual events with the theoretical distribution implied by the uniform distribution and independence assumptions. The standard chi square test was used as a basis of evaluation.

The following steps were used to evaluate the actual distribution of events for all tests.

1. Set up intervals associated with equal probabilities of occurrence. (The theoretical distribution is easy to calculate since each interval then has equal expected observations.)

2. Observe a series of events.

3. Count the number of events falling in each interval.

4. Use a chi square test to determine whether the observed number of events falling in each interval accords with the number theoretically expected to fall in those intervals under the assumption of independence.

The tests and a brief explanation of each are listed below:

Tests of Uniformity

These were the most straightforward tests conducted. The random numbers generated were treated as samples of a random variable. The hypothesis tested was that the random variable was uniformly distributed over the unit interval between zero and one.

Test for Uniformity of Singles (Test 1). The unit interval between 0 and +1 was divided into 100 equal sub-intervals. For each of the sub-intervals, a count was obtained of the number of generated random numbers falling in that interval. The chi square test involved testing whether these observed numbers of cases falling in the sub-interval deviated significantly from the numbers of cases theoretically expected to fall in the sub-intervals on the average under the hypothesis that the random numbers generated behave as a random sample of a uniformly distributed random variable.

Test for Uniformity of Pairs (Test 2). The unit square was divided by a 10 x 10 grid into 100 equal square subdivisions. Each pair (u_i, u_{i+1}) of random numbers generated was then assigned to one of the square subdivisions according to its coordinates. The numbers of random numbers falling in the respective square subdivisions were compared with the

corresponding numbers of cases theoretically expected to fall in these subdivisions under the hypothesis that the successive numbers generated were samples from two independent and uniformly distributed random variables.

Test for Uniformity of Triples (Test 3). The unit cube was divided by a three-dimensional grid into 1000 = 10 x 10 x 10 cubic subdivisions. Each triple (u_i, u_{i+1}, u_{1+2}) of random numbers generated was assigned to one of the cubic subdivisions according to its coordinates. The observed numbers of cases falling in these sub-intervals were compared with the theoretically expected numbers of cases that would fall in the subdivisions under the hypothesis that the successive numbers generated were samples of three independent uniformly distributed random variables.

Maximum Number in a Subset

Let X be a continuous random variable uniformly distributed on the unit interval (0, 1). Then $P(X \leq x) = x$. Consider a random sample of n observations of X. Then the probability that a sample will contain no value greater than x_{max} is

$$P\left[\bigcap_{i=1}^{n} (X \leq x_{max})\right] = \prod_{i=1}^{n} (P(X \leq x_{max})) = x_{max}^{n}$$

Consider now the random variable $Y = X_{max}$ where X_{max} is the maximum value in a sample of n observations of the random variable X. Then

$$P[Y \leq y] = y^n$$

describes a cumulative distribution function for Y. Since $0 \leq Y \leq 1$ for all Y, the probability can be considered that an observed value of Y will fall in the sub-interval (a,b) of the unit interval to be

$$P[a < Y \leq b] = P[Y \leq b] - P[Y \leq a] = b^n - a^n.$$

In N observations of the random variable $Y = X_{max}$, $(b^n - a^n)N$ cases of Y on the average would be expected to fall in the interval (a,b).

Since the above development crucially depends on the independence of the observations of the random variable X, the probability distribution function of the maximum value in a sample of n observations may be used as a basis for testing the independence and uniformity properties of a random number generator. In the present study, the unit interval was divided into sub-intervals associated with equal probabilities of containing specified values of $Y = X_{max}$. Intervals associated with equal probabilities were chosen to facilitate the accuracy of the chi square tests of the random number generator. The following tests were performed:

Maximum of Two (Test 4). The unit interval was divided into 25 sub-intervals associated with equal probabilities of occurrence, and the maximum value of a pair of random numbers u_i, u_{i+1} was assigned to the respective sub-interval. The resulting number of observations for the chi square test was equal to one-half the sample size.

Maximum of Five (Test 5). The unit interval was divided into 20 intervals. The maximum value of the observed sequence u_i, ..., u_{i+4} was assigned to the appropriate sub-interval. The number of observations was equal to one-fifth the sample size.

Maximum of Ten (Test 6). The unit interval was divided into 10 sub-intervals. The maximum value of the sequence u_i, ..., u_{i+9} was assigned to the appropriate sub-interval. The number of observations was equal to one-tenth the sample size.

Minimum Number in a Subset of n Numbers

By an argument analogous to that developed in connection with the probability of occurrence of maximum values in a sequence of n observations of a uniformly distributed random variable, the probabilities associated with the minimum values for such sequences can be developed. Specifically, let X_{min} be the minimum value of n independent observations of a uniformly distributed random variable X. Then the random variable $Z - X_{min}$ is distributed such that

$$P[Z > z] = P[X_{min} > x_{min}] = (1 - x_{min})^n$$

$$\text{or} \quad P[Z \leq z] = 1 - (1 - x_{min})^n.$$

From this distribution function the probability that an observed value of Z falls in a specified sub-interval (a,b) on the unit interval (0,1) can be computed by

$$P[a < Z \leq b] = P[Y \leq b] - P[Y \leq a] = (1 - a)^n - (1 - b)^n$$

The following tests for minimums were made on the sample of random numbers generated. These tests were carried out fully in analogy to the tests for maximums:

Minimum of Three (Test 7). Each of the first three sets of three numbers from each set of ten was considered and the tenth number omitted, with the resulting number of observations equal to one-third the sample size.

Minimum of First Three (Test 8). Only the first three numbers from each set of ten were considered, resulting in a total number of observations equal to one-tenth of the sample.

Minimum of Seven (Test 9). Identical to Test 8, except that the first seven numbers in each set of ten were considered.

Minimum of Ten (Test 10). Each set of ten numbers was considered.

Sum of n Numbers

Let S_n be the statistic $S_n = X_1 + ... X_n$, the sum of n independent random variables X_1, ..., X_n uniformly distributed over the unit interval (0,1). According to Parzen (4),

$$F_n(S) = P[S_n \leq S] = \frac{1}{n!}\left(S^n - \frac{n!}{(n-1)!1!}(S-1)^n + \frac{n!}{(n-2)!2!}(S-2)^n - \frac{n!}{(n-3)!3!}(S-3)^n + ...\right)$$

if $0 \leq S \leq n$; $P[S_n < S] = 0$ if $f < 0$; and $P[S_n \leq S] = 1$ if $S > n$. The series is summed until a term within the parentheses, $\frac{n!}{(n-j)!j!}(S-j)^n$, is encountered such that $j > n$ or $j > S$. This formula can be used to determine the probability of encountering sums of independent uniformly distributed random variables in any sub-interval (a,b) of the interval (0,n) by $P[a < S_n \leq b] = F_n(b) - F_n(a)$.

Since the above development again depends on the independence and uniformity of identically distributed random variables, it can form a basis for testing a random number generator. By observing the relative frequencies with which certain sub-intervals of the interval (0,n) contain observed sums of n sequentially generated uniformly distributed random numbers, we can determine by a chi square test whether these frequencies accord with their theoretically determined expected values under the assumptions of independence and uniformity. The following sums tests were used:

Sum of Two (Test 11). Each pair was considered, resulting in a total number of observations equal to one-half the sample size.

Sum of Four (Test 12). The first four from each set of five numbers were considered. (The fifth and tenth numbers in each set of ten were omitted.) The total number of observations was equal to one-fifth of the sample size.

Sum of Seven (Test 13). The first seven numbers from each set of ten were included, resulting in a total number of observations equal to one-tenth of the sample.

Sum of Ten (Test 14). This test was identical to Test 13 except that ten numbers were used.

Fourth Largest of Five (Test 13)

Let $X_1, X_2, \ldots, X_5$ be five independent random variables uniformly distributed on the unit interval (0,1). Let G be the statistic G = (the fourth largest of $X_1, X_2, \ldots X_5$). From Rao (5), the cumulative distribution function for exactly three numbers larger than g and one number smaller than g in a group of 5 numbers from 5 independent trials is

$$F_G(g) = \frac{5!}{3!1!1!} \int_o^g (1-g)^3 \, g \, dg$$

$$F_G(g) = 10g^2 - 20g^3 + 15g^4 - 4g^5$$

The unit interval may be divided into sub-intervals each having equal probability of containing sample values of G. The probability that G will fall in the interval (a,b) is given by

$$P[a < G \leq b] = F_G(b) - F_G(a)$$

This fact may be used to test the independence and uniformity properties of a uniform random number generator. In the present study, the unit interval was divided into 10 intervals each having equal probabilities of containing sample values of G. A chi square test was run on the observed frequencies in these intervals based on the first five random numbers of each sequence of 10 random numbers generated. The total number of cases observed was equal to one-tenth the number of random numbers generated in the total sample.

Each set of ten sequentially generated random numbers was ordered from smallest to largest and the difference between each adjacent pair was calculated, including the difference between the smallest number and zero. Assignment was made to the interval containing the value of the maximum differences. The total number of events was equal to one-tenth of the sample.

Maximum Difference for a Set of Order Statistics

Let $x_1, \ldots, x_n$ be a set of joint observations on a set of random variables $x_1, \ldots, x_n$ uniformly distributed on the unit interval. Consider that the n observations are then rank ordered from smallest to largest. By $X_{(1)}$, denote the random variable "smallest of n numbers"; by $X_{(2)}$, denote the random variable "next to smallest of n numbers"; and so on, to $X_{(n-1)}$, "the next to largest"' and $X_{(n)}$ "the largest of n numbers". Now, define the statistic $V = (U_1, U_2, \ldots, U_n)$ where $(U_1, U_2, \ldots, U_n)$ corresponds to $(X_{(1)}, (X_{(2)} - X_{(1)}), \ldots, (X_{(n)} - X_{(n-1)})$. The cumulative distribution function

$$F(v) = P[V \leq v] = \sum_{k=0}^{m} (-1)^k \binom{n}{k} (1 - KV)^n$$

(where m is an integer such that $\frac{1}{m+1} < V \leq \frac{1}{m}$) gives the probability of obtaining a maximum difference of order statistics no larger than V.

Maximum Difference for Each Ordered Set of 10 Numbers (Test 16). Each set of ten sequentially generated random numbers was ordered from smallest to largest and the difference between each adjacent pair was calculated, including the difference between the smallest number and zero. Assignment was made to the interval containing the value of the maximum difference. The total number of events was equal to one-tenth of the sample.

Autocorrelation between the jth and jth plus ten numbers (Test 17). The correlation coefficient between the jth and jth plus ten numbers was calculated for the entire sample using the formula

$$r = \left(\sum_{i=1}^{N-10} (x_i \, x_{i+10}) - \left(\sum_{1}^{N-10} x_i \right)^2 \right) \Big/ \left(\sum_{1}^{N-10} x_i^2 - \left(\sum_{1}^{N-10} x_i \right)^2 \right)$$

where r represents the correlation coefficient, and N = total sample size. This test was evaluated by considering the probability that a set of n pairs would have an r as large or as small as the observed if the correlation were in fact equal to zero. Although there was only one event per sample, we were still able to compare an actual distribution to a theoretical distribution when evaluating the 200 samples of 1000.

Runs Up and Down (Test 18). The number and lengths of runs were counted and assigned to the corresponding interval, forming a frequency distribution. The theoretical distribution was determined by the formula

$$E(r_p) = \frac{2*n\,(p^2 + 3p + 1) - 2(p^3 + 3p^2 - p - 4)}{(p + 3)!}$$

where p represents the length of the run, r_p represents the number of runs of length p, n the sample size, and $E(r_p)$ the expected number of such runs.

Runs Above and Below the Mean (Test 19). The number and length of runs above and below the mean were counted and an assignment was made to the corresponding interval in the domain of lengths of runs. The expected value of the number of runs of length p was given as

$$E(r_p) < \frac{(n - p + 3)}{2^{p+1}}$$

A chi square test was used to confirm the reasonableness of actual results.

RESULTS OF STATISTICAL TESTS

Output of One Generator

The results of the initial tests conducted on the first sequence are summarized in Table 1. These tests, conducted on sets of ten as described in the preceding section, involved a total sequence of 200,000 consecutive numbers (beginning with an initial value of $x_o = 1$). For each test, there were ten chi square values based on the distributions resulting from 2000 sets of ten numbers and an overall chi square value based on the other ten statistics. The column headings are as follows:

a. Test (indicates the name and number of the test as listed in the section describing the statistical tests).

b. Number of intervals used.

c. Number of sample points (for each sub-sequence of 20,000 numbers).

d. Expected observations per interval [(c) ÷ (d)].

e. Minimum chi square value (expressed as a standard deviation[4]) of the ten values based on the distributions resulting from 2000 sets of ten numbers.

f. Probability that the smallest statistic of ten independent values would be no smaller than the minimum.

g. Maximum chi square value (expressed as a standard deviation[4]) of the ten values based on the distributions resulting from 2000 sets of ten numbers.

h. Probability that the largest of ten independent values would be no larger than the maximum value observed.

i. Overall chi square statistic (expressed as a standard deviation[4]) based on the ten chi square values.

j. Probability of a chi square statistic as large as that expressed by (i).

---

[4] McNemar, Quinn. Psychological Statistics, p. 197. For n's larger than 30, the expression $\sqrt{2\chi^2} - \sqrt{2n - 1}$ will have a sampling distribution which will follow very closely the unit normal curve.

In evaluating the chi square statistics, both the low and high values are of interest. Excessively low values imply that the distribution for a particular test is not sufficiently random--that it is too uniform, while excessively high values imply that the distribution is not sufficiently uniform. In addition to the individual chi square values, the distribution of chi square values for a particular test, which the probabilities in (f) and (h) are designed to evaluate, is of concern. The tests which fail at the 5 percent and 95 percent confidence levels are circled in Table 1. Overall, the performance was reasonably satisfactory. If the tests were independent (which they are not), there would be expected 10 percent (5 percent at each tail) or approximately 5 failures, whereas actually there were eight failures.

The minimum chi square value out of ten values based on the distribution of the sums of four (Test 12, column e) is somewhat disappointing. The expectation that 10 tests would yield no chi square statistic smaller than that observed (.39 standard deviation below the mean) would be 1 percent. In other words, 99 percent of the time at least one distribution out of ten using 100 degrees of freedom would have a smaller chi square value than 93.1. Similarly, the test for the maximum of five (Test 5, column g) was not completely satisfactory. Only 1 percent of the time would the largest of ten chi square values be only .36 standard deviations above the mean.

The results of the tests conducted on the first eight sequences are summarized in Table 2. These tests were not designed to test individual chi square values for each sequence and test, but rather to evaluate the distribution of chi square values to determine if it conformed to the theoretical distribution.

For each sequence, the full complement of tests previously described was conducted on the first 200,000 numbers, each test being applied to 100 sets of ten (i.e., 200 samples of 1000). This yielded 200 chi square statistics for each sequence and test. These statistics were then arrayed in a frequency distribution of twelve intervals based on both the degrees of freedom for the particular test and Table A-6b, "Percentiles of the X^2/df Distributions", in Dixon and Massey (6).

Two intervals at each end had an expected observation of 5 percent (10) and the eight intervals in the middle had an expected observation of 10 percent (20). Finally, a chi square test was made on each of 152 distributions (8 sequences x 19 tests); Table 2 indicates the chi square value divided by degree of freedom (11) for each sequence and test.

Table 2 has eight columns, each representing a sequence. The sequence number and multiplier (c) head each column. At the left are the name of the test and, where appropriate, the number of intervals and total observations per sample of 1000. In addition, the expected observations per interval have been indicated for those tests which have an equal number of expected observations per interval. Within the table are the chi square divided by degrees of freedom values, as described above, corresponding to each sequence.

Table 1

RESULTS OF INITIAL TESTS ON FIRST SEQUENCE OF SETS OF TEN

(a)		(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)
Numbers	(1)	100	20,000	200	-1.54	.53	.92	.14	-.34	.63
Pairs	(2)	100	10,000	100	-.53	.03	1.00	.18	.84	.20
Triples	(3)	1000	6,000	6	-1.95	.77	1.01	.18	-1.82	.97
Max-two	(4)	100	10,000	100	-.76	.08	1.88	.74	1.65	.05
Max-five	(5)	100	4,000	40	-1.45	.47	(.36)[a]	.01	-.75	.77
Max-ten	(6)	100	2,000	20	-1.81	.70	1.33	.38	-.61	.73
Min-three	(7)	100	6,000	60	-2.50	.94	1.86	.11	-.30	.62
Min-first three	(8)	100	2,000	20	-.91	.14	1.59	.56	.48	.32
Min-last seven	(9)	100	2,000	20	-1.46	.47	1.48	.49	-.51	.70
Min-ten	(10)	100	2,000	20	-1.87	.73	1.55	.54	.88	.19
Sum of Two	(11)	100	10,000	100	-1.40	.43	1.84	.72	-.81	.79
Sum of Four	(12)	100	4,000	40	(-.39)	.01	1.17	.28	1.89	.03
Sum of Seven	(13)	100	2,000	20	-1.94	.77	1.73	.65	-.49	.69
Sum of Ten	(14)	100	2,000	20	-1.78	.[illegible]8	1.08	.22	-.15	.56
Fourth Largest of Five	(15)	100	2,000	20	-1.78	.68	1.67	.61	-.47	.68
Max Diff Set	(16)	100	2,000	20	-.64	.05	2.13	.85	1.58	.06

$x_o = 1$

c = 13501 3333

[a] Tests which fail at the 5% and 95% level of confidence have been circled.

The tests which fail at the 5 percent and 95 percent confidence levels have been circled. Overall performance appears satisfactory. For example, it was expected that 7 or 8 of 152 independent tests would fail at the 5 percent level, and in fact, six tests yielded values below that level (.42). At the 95 percent confidence level, 7 or 8 failures were also expected, and fifteen had values above that level (1.79). Although the number of failures at this level is somewhat high, the amount is not unreasonable, particularly since all tests are not completely independent.

Several of the tests did not have completely satisfactory results when considered individually. For example, the pairs test failed at the 95 percent level for both the fourth and seventh sequences. There is slightly less than a 5 percent chance that two out of eight independent tests will fail at the 95 percent level. Similarly, the sum of seven tests failed for both the third and fourth sequences at the 95 percent level.

In spite of the failures mentioned above, the generator appears to yield distributions with satisfactory statistical properties. For all tests, the chi square values based on the samples of 1000 cover the full range of expected values. Although in some cases the actual distribution may not approximate the theoretical distribution as closely as desired, the range and distribution of values seems sufficient for present needs.

Some additional comments are in order concerning the runs tests, particularly the tests for runs above and below the mean. In evaluating the distributions of runs above and below the mean, a loss of two degrees of freedom was assumed. This assumption was made because 1) the number of runs above the mean must be within one of the number of runs below the mean, and 2) the sum $(a_1 + b_1)*1 + (a_2 + b_2)*2 + \ldots + (a_n + b_n)*n$, where a_i stands for the number of runs of length i above the mean and b_i stands for the number of runs of length i below the mean for i = 1,n, must equal 1000.

However, it was necessary to group runs of length five or more above the mean into one cell and similarly to group runs of length five or more below the mean to avoid having fewer than ten expected observations per cell; thus, the formulae indicated above do not literally hold. Since the number of runs above the mean must be within one of the number of runs below the mean and since some runs were grouped, we are probably not losing a full two degree of freedom, but something less than that. It can be hypothesized that since we may have used a smaller value for the degrees of freedom than actually existed and since the chi square statistics were evaluated by dividing by the df, there may have been a slight bias toward the high end of the distribution. This hypothesis is consistent with all the actual distributions, each of which had more high values than expected. When the same tests were applied to the same sequences of numbers, but in blocks of 100,000 and with 24 cells (12 for runs above and 12 for runs below), the chi square values were completely acceptable. This result is also consistent with the hypothesis, since any bias existing for the reasons outlined above is substantially reduced as the number of cells is increased.

Table 2

CHI SQUARE VALUE DIVIDED BY DEGREES OF FREEDOM FOR EACH SEQUENCE AND TEST

X_o = 0203050704020605 For All Sequences Multiplier: c				(1) 135013333	(2) 140014333	(3) 143015333	(4) 146016333	(5) 151017333	(6) 154020333	(7) 157021333	(8) 162022333
Test	Number of Intervals	Number of Sample Points	Expected Observations Per Interval								
Singles (1)	50	1000	20	1.33	.80	1.31	.74	1.05	.70	1.28	.77
Pairs (2)	25	500	20	.50	(.42)[a]	1.05	(2.00)	.84	.63	(1.92)	1.18
Triples (3)	27	297	11	1.17	1.44	1.16	.92	(.40)	(1.88)	1.06	1.10
Maximum of Two (4)	25	500	20	(1.85)	1.02	.88	.71	.73	1.25	1.01	1.00
Maximum of Five (5)	20	200	10	1.71	1.30	.79	1.78	.62	.69	1.18	(.42)
Maximum of Ten (6)	10	100	10	(.40)	.77	.81	.75	1.11	.70	1.34	.94
Minimum of Three (7)	20	300	15	(2.04)	.46	.58	.53	1.15	1.59	.93	.92
Minimum of First Three (8)	10	100	10	1.31	1.36	.79	1.03	.45	1.03	.66	.55
Minimum of Seven (9)	10	100	10	.73	1.35	.65	1.24	.62	(.40)	.57	.81
Minimum of Ten (10)	10	100	10	.88	1.49	1.25	.95	1.16	1.32	.85	.65
Sum of Two (11)	25	500	20	1.48	.94	.69	.53	.60	1.63	.72	1.56
Sum of Four (12)	20	200	10	.89	(1.93)	.98	.60	1.71	1.00	1.19	1.27
Sum of Seven (13)	10	100	10	.47	1.45	(2.09)	(2.02)	.74	(.39)	.71	1.25
Sum of Ten (14)	10	100	10	1.21	1.72	1.00	(2.10)	1.33	1.23	.88	.91
Fourth Largest of Five (15)	10	100	10	.89	.51	(1.80)	.62	.71	1.22	(1.93)	.67
Maximum Difference (16)	10	100	10	.49	.71	1.15	.55	.86	1.09	.45	1.23
Set of Order Statistics Runs Above and Below Distribution (17)	10	-	-	1.56	1.47	1.67	(1.99)	(2.23)	1.54	1.55	1.00
Runs Up and Down Distribution (18)	8	-	-	1.06	1.00	1.02	.48	(1.80)	.85	.64	.98
Autocorrelation between J and J+10 (19)	-	-	-	1.27	1.34	.80	1.07	.70	.96	(1.82)	1.38

[a]Tests which fail at the 5% and 95% level of confidence have been circled.

Relationship between Sequences from Different Generators

The tests previously outlined and described were designed to evaluate characteristics of the numbers output by a single generator. Frequent application to problems utilizing simultaneously sequences generated by several generators requires evaluation of the relationship between such sequences. The following tests were used for this purpose.

Correlation. This test involved calculating the correlation between all combinations of pairs of twenty multiplicative generators, each having a different initial value and multiplier, using the same formula used for the autocorrelation tests. The test was conducted on 10 sets of 1000 numbers from each generator, yielding 10 correlation coefficients for each pair of generators. The results of these tests are summarized in Table 3. Only the correlation coefficients based on the first eight generators are reported. The distribution of coefficients yielded a chi square value of 7.28, which falls at approximately the 25th percentile for 11 degrees of freedom.

Table 3

DISTRIBUTION OF CORRELATION COEFFICIENTS AMONG EIGHT MULTIPLICATIVE GENERATORS

$\bar{R}$	P	Expected	Actual
-.052	.05	14	12
-.041	.10	14	12
-.027	.20	28	26
-.017	.30	28	29
-.008	.40	28	35
0	.50	28	30
.008	.60	28	24
.017	.70	28	30
.027	.80	28	21
.091	.90	28	33
.052	.95	14	11
-	1.00	19	17

Rank order across generators. This test was designed to evaluate the rank of the numbers produced by each generator relative to the numbers generated by the other generators. A number was generated for each of the 8 generators, ranking the 8 numbers from highest to lowest, counting the

numbers of occurrences in each of the 8 ordinal positions for each generator, and then making a chi square evaluation of the distribution for each generator (8 cells) and over the 8 generators. The null hypothesis of independence would be rejected if some ranks were more frequently associated with some generators than with others.

Table 4 summarizes the results. The first column indicates the generator and the second column the standard deviation of all chi square values for the generator. This value was calculated by summing the 20 chi square values computed for that generator and using the standard formula, $\sqrt{2X^2} - \sqrt{2n - 1}$ with n = 140 (20 tests times 7 degrees of freedom). The last column indicates the probabilities of getting an overall chi square value as low as that observed. All values were well within an acceptable range, although the final sequence gave overall chi square that would be exceeded more than 90% of the time.

Table 4

RANK ORDER ACROSS SEQUENCES

Generator	Standard Deviation	Probability
1	-.71	.24
2	+.63	.73
3	-.80	.21
4	-.19	.43
5	+.42	.66
6	+.45	.67
7	-.87	.19
8	-1.33	.09

Uniformity tests. In addition to the correlation and rank order tests across sequences, uniformity tests were conducted for pairs and triples. These tests were identical to the uniformity tests already discussed except that the pairs were formed by selecting one number from each of two generators and the triples consisted of a number from each of three generators. These tests were also completely satisfactory. There appears to be no unacceptable relationship between any of the first eight generators. Similar tests among subsequent generators have not been made, but there seems to be no reason to suggest that results would be different from those obtained with the eight tested.

The generators described here have been in use at BESRL since mid-year 1968.

LITERATURE CITED

1. Niehl, Elizabeth and Richard C. Sorenson. SIMPO-I entity model for determining the qualitative impact of personnel policies. BESRL Technical Research Note 193. January 1968.

2. Coveyou, R. R. and R. D. MacPherson. Fourier analysis of uniform random number generators, Journal of the Association for Computing Machinery, 14, 1 (January 1967), pp. 100-119.

3. Marsaglia, C. Random numbers fall mainly in the planes. Proceedings National Academy of Science, 60,5 (September 1968).

4. Parzen, Emanuel. Modern Probability Theory and its Applications. John Wiley and Sons, Inc., New York, 1960.

5. Rao, C. Radharkrisha. Advanced Statistical Methods in Biometric Research. John Wiley and Sons, Inc., New York, 1952.

6. Dixon, Wilfred J. and Frank J. Massey, Jr. Introduction to Statistical Analysis, 2d Edition. McGraw Hill Book Co., New York, 1957.

SELECTED BIBLIOGRAPHY

Birnbaum, Z. W. Introduction to Probability and Mathematical Statistics. Harper and Brothers, New York, 1962.

Fisz, Marek. Probability Theory and Mathematical Statistics. John Wiley and Sons, Inc., New York, 1963.

Lehmer, D. H. Mathematical methods in large-scale computing units. Proceedings of Symposium on large-scale digital calculating machinery. Harvard University Press, 1949.

MacLaren, M. D. and G. Marsaglin. Uniform random number generators. Journal of the Association for Computing Machinery, 12.1, p. 83, 1965.

Neumann, J. von. Various techniques used in connection with random digits. N.B.S. Applied Mathematics Series, No. 12, 1951.

Peach, Paul. Bias in pseudo-random numbers. Journal of the American Statistical Association, 56:295, p. 610, 1961.

Tocher, K. D. The Art of Simulation. The English Universities Press Ltd., London, 1963.

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*

U. S. Army Behavioral Science Research Laboratory, Arlington, Virginia

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

EVALUATION OF A MULTIPLICATIVE GENERATOR OF PSEUDO-RANDOM NUMBERS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

Frederick W. Kuehl

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
September 1969	32	6

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO. DA R&D Proj. No. 2Q025101M711 c. SIMPO d. d-12	Technical Research Note 215 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Office, Chief of Research and Development, DA

13. ABSTRACT

The present study reports on the development and testing of a pseudo-random number generator which would be able to rapidly supply the SIMPO-I entity models with numbers meeting essential tests for random qualities. The generator developed by BESRL is a power residue or multiplicative generator, the simplest of the many generators based on the linear congruential method. In addition to an acceptable generator, the work accomplished resulted in a compilation of statistical tests which have wider application for evaluating generators used on other computer systems. The tests are described, and essential mathematical formulations are presented.

DD FORM 1 NOV 68 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
*Simulation						
*Simulation model						
Computerized models						
*SIMPO-I entity models						
Statistical tests						
*Pseudo-random number generator						
*Random numbers						
Computer systems						
Binary numbers						
*Multiplicative generator						
Sampling techniques						
Computer routines						
Probability distribution theory						
*Mathematical operation						